Preventing Accidental Privacy Leakage In Ubiquitous Visual Sensing

by

Ying Zou

A thesis submitted in partial fulfillment
of the requirements for the degree of
Master of Science
(Computer and Information Science)
in The University of Michigan-Dearborn
2017

Master's Thesis Committee:

Associate Professor Di Ma, Chair
Associate Professor Jinhua Guo
Associate Professor Shengquan Wang

Table of Contents

List of Figures

List of Tables

Abstract

The digital cameras are ubiquitous in our daily life and became an essential part of everyday devices. For example, there are cameras in cellphones, tablets, and many other wearable devices. This facilitates capturing and storing information around us through taking images and videos. Despite of the easiness and usefulness of cameras, they also bring a lot of serious privacy problems. These privacy problems can be either malicious intent or accidental user own ignorance, for example, malicious secret filming or unwanted recognition from a photo. To illustrate, the camera cannot distinguish whether part of the data captured is private or sensitive. Once the picture is exposed, accidental privacy leakage may happen.

In many cases, we cannot avoid the malicious picture capture. However, we can avoid the information leakage in proactive way. Our goal in this thesis work is to prevent or minimize accidental privacy leakage. To achieve it, the basic idea is to mark the sensitive areas or objects by QR code, in order to encode the privacy information in such targets. Thus those areas and objects can be filtered out before publishing. Usually, face is a common case as the sensitive information, which we used in this thesis. In our implementation, the recognized faces were blurred from the original picture before publishing.

We discussed the implementation of proactive protection of information from three aspects, including real time none-object based approach, offline none-object based approach and object based approach.

In the real-time approach, QR code is used to implement real time processing. By using this feasible method, user can have fine grained control on what is revealed and what is kept private.

In the offline method, a range selector is implemented to eliminate the specific area, even though there are still some defects of this approach. For example, it is not real time, and the lists of

sensitive area differs from picture to picture. To help our implementation, this research utilized face recognition for the object based method as well. Compared with none object based sensing, this approach has less feasibility, lower speed, and less accuracy.

The thesis categorizes existed ways by two types (permission requirement and predefine sensitive area) and compares these ways from different perspectives through showing the pros and cons for each side.

Chapter 1 Introduction

1.1 Background

Ubiquitous sensing is used to enhance the sense of the physical and social contexts of our daily life, in order to provide continuous services and support our life quality.

The proliferation of embedded cellphone and wireless sensors like google glass, Galaxy Gear, and applications based on these platforms enable an easy way to access cameras, capture data and process it, however, it is hard to distinguish whether the data is captured by the camera is private or sensitive, in this manner the risk of information leakage increases as well.

Nowadays many applications are installed on the ubiquitous mobile devices at the same time. That is the platform may run multiple software applications simultaneously. In addition, the capability of communication with other devices changes rapidly. More and more devices use such as Wi-Fi, Bluetooth, RFC and so on, which leads more dangers than before to ensure the information security during wire or wireless communications.

The concepts as isolation and safe communication are required to be implemented in the platform. For instance, in order to ensure the safety in the whole procedure, the application and data collection sensors (e.g. camera) can be isolated or separated by using a framework or other mid-layer software; authorization, authentication and encryption/decryption can also be used in communication establishment.

In general, the malicious leakage like the hacker attack or intentional secret filming will not be discussed in this thesis. Since the boundary of malicious leakage is hard to detect and define, as

well as the solution is usually based on specific case, it is difficult to find a general method to solve all cases. The accidental leakage and the solution to prevent the accidental leakage will be focused in this thesis. Accidental leakage means the information is utilized by improper usage, rather than private information which is leaked by the user's intention.

1.2 Motivation

The researchers found that the advertisements of application leak millions of sensitive personal information of mobile phone users including how much money we have, whether we have children, and our political preferences [18].

For example, many people like to use cell phone to take pictures during their daily life. They take pictures for their homes and their families, and then post them on social network website like Twitter, Facebook and so on. People or software may analyze these pictures to collect the private information [23].

For example, the accidental password leakage in 2014 world cup, the Wi-Fi SSID and password for the security center were exposed by a photograph of host nation Brazil's federal police occasionally, which captured with the content written on a white board. In the background of such picture, it is possible to read the SSID "WORLD CUP" and password "b5a2112014" easily [21]. Same issue also happened in 2014 super bowl, Wi-Fi information includes the password was leaked in the live broadcast [22].

To illustrate more, we always see the Wi-Fi with password in many public places, it may be a specific benefit for customer who had dinner or shopped in these places. A recent Xirrus survey found that 91 percent were aware of public Wi-Fi security risks, 89 percent ignored them and connected anyway [24]. Therefore, hiding the Wi-Fi password before the selfie can help the owner and user to protect the security of Wi-Fi use.

For instance, some may know your social relationship without your permission if the face of your friend or family is exposed. So sharing the picture after eliminating sensitive information can minimize and even eliminate many problems.

In the scenario of a presentation, for instance, the business representative is demonstrating the newest production. The content in the whiteboard includes some sensitive information that should not be exposed, while the audience may want to use their cameras to record the important items for further review. Seems in this dilemma, if the audiences want to avoid the information leakage, they should follow the instructions to capture the image with the sensitive information extracted.

To date, there are two approaches in existed models of the privacy protection. The first one is called object detection, in which the application / framework can ask user to provide the permission when any potential information leakage was detected automatically. And then the application / frame could continue to retrieve the content of image only after permission is acquired.

In the implementation of the recognizer in Enabling Fine-Grained Permissions for Augmented Reality-Applications with Recognizers [5], a recognizer uses the raw data from sensor as its input and extracts higher-level objects (e.g. face), to applications. They tried to propose a fine-grained permission system when applications request permissions of specific objects.

This approach may work well with well-defined sensitive objects. If sensitive objects cannot be well-defined, the object detection based approach can no longer work. For instance, detecting a ceiling, it is difficult to define what is ceiling, because there is no obvious characteristic of ceiling. Therefore, for some objects, the accuracy of detection cannot be guaranteed.

Objective detection is also not feasible if the software can't learn the model of appearance recognition accurately. Some algorithms try to improve the solutions even the objects are distorted, drifted or rotated, such as Tracking-Learning-Detection (TLD) [7], which combines tracking, learning and detecting to follow the objects appears in the video.

Another method is predefining the sensitive area, and it is a non-object related approach. In this method, the user will predefine the sensitive area or filter, which will be reserved by a software or platform for further operations. The further operation could be erasing or blurring privacy information in filter or labeling the information by privilege. However, the current implementations are infeasible. It contains limited list of sensitive filters, and for different scenarios, the list may be different too.

In order to evaluate the approaches raised in this thesis, such three methods in sensitive information protection will be applied and compared in this thesis.

With regard to the first method, the face detection by OpenCV will be used in this thesis work to show the concept of object based leakage preventing, and as mentioned above, the disadvantage is apparent. Currently the objects need to be clearly defined and confined (e.g.: face, lane, car etc.). Comparing with none object based sensing, this method had less feasibility because the specific objects need to be predefined, and come with lower speed, also lower accuracies. However, lots of implementations for object detection are used broadly in reality. We will improve the performance of detection in future work.

The second method is QR code based leakage preventing, in which the QR code in picuture was used to pre-define sensitive area, and then to implement real time processing. The user can mark the picture with this QR code that includes coordinate x, y length, width and the access level. By using this method, the user can take fine grained control on what is revealed and what is kept private.

Since the filter lists is predefined, they can be used continuously till the lists are changed, the procedure is repeatable. But the accuracy of recognition is dependent on the gratuity and clearance of QR code, thus we have to tradeoff either efficiency or recognition accuracy due to such disadvantages.

The last but not least one, is offline leakage preventing. The detection of QR code and image process are separated by two. For this method, QR code detector is implemented independently.

After the detection has been completed successfully, the result of detection will be reserved for the picture processor. The advantage of this method is easy to implement and come with high efficiency, while it's not real time as an obvious disadvantage, and it also increases the operations for users.

Chapter 2 Related Work

2.1 Object Detection

For the method of object detection, as illustrated above, the software/application detects the possible object, and then asks for permission. Application/frame will continue to retrieve the content of image only after permission is acquired.

Scanner Darkly is a privacy middle layer for third-party perceptual application. It can reduce the privacy leakage, and can also increase the accuracy and functionality of these applications. Scanner Darkly integrated OpenCV into its implementation, and meanwhile, access control, algorithmic privacy transforms, and user audit are all included in Scanner Darkly [6].

As mentioned above, it is needed to tradeoff between the accuracy of detection and user privacy. Sensor sift provides an optimized algorithm to improve this concern [4]. It will minimally expose the private properties defined by customer, and maximize the exposure of public properties predefined.

Deep learning is also utilized in object detection usually. Deep learning refers to feather abstraction of multiple layers, the set of algorithms that extract or subsample the feathers from lower layer, while lower layer learned these features from row data. Researcher from Stanford University used the well-known datasets to train and test CNN (convolution neural network), and they achieved good result in object detection [20].

This implementation can also be extended to video processing, the proposed framework resolved the problem of detecting privacy sensitive information by utilizing a proactive human object detector, then blurring or blocking out the privacy sensitive information [19]

2.2 Pre-defined Sensitive Areas

Another method is to predefine sensitive area, its non-object related. Users predefine the sensitive area or filter, software or platform reserve these filters for further operations.

There is an interesting approach to detect the object by using predefined colors. For example, the user wants to hide one's face who wears colored hats or vests as predefined, the system will automatically track these hats or vests to identify the location and size of each face, and then blurs faces with solid overlays, while minimizing the covered zone and maximizing the remaining area of the picture.

Jeremy Schiff and his colleagues and students from the University of California Berkeley, also proposed an approach to solve the privacy issue brought by digital camera. The approach deduced the location and the size of each object by using statistical learning and classification technologies, and it can be applied in both static image and dynamic video by using AdaBoost classifier and SIR filter. Still this solution requires predefining the colored marker to enhance the recognition of objects [9].

One solution is named world-driven access control, where the object needs to present themselves to the OS. They implemented this approach by using the technology like passport. The passport is used to avoid the leakage of sensitive information, also provide a method to configure the access control.

Robert, Mohammed, David and Apu from the Indiana University Bloomington proposed an amazing technique named Place Avoider, which can recognize the sensitive environment and then flag the images or videos captured in such places, to hide the private information before showing to the application or end user, by using both scene-level image features and fine-grained features to indicate the location where the screen captured. In this solution, predefined features such as specific object, color or texture will be used to filter the sensitive images [17].

With regards to the method of using QR code for predefining area, the accuracy of recognition cannot be guaranteed, if QR code only occupies a small area in the picture. Another improved algorithm was implemented in Fast Component-Based QR Code Detection in Arbitrarily Acquired Images [3].

Chapter 3 Problem Definition and Proposed Approach

3.1 Problem Definition

There are two approaches in existed models of the privacy protection. For all approaches, the image processing after detection will use the same technology AES to do encryption and decryption. So the main difference is in the detection of sensitive area.

One is object-based detection, in which the application / framework detects instances of semantic objects of a certain class (such as faces, pedestrian, or vehicles) in digital images and videos.

The example of object (face) detection is showing as figure 1 below.

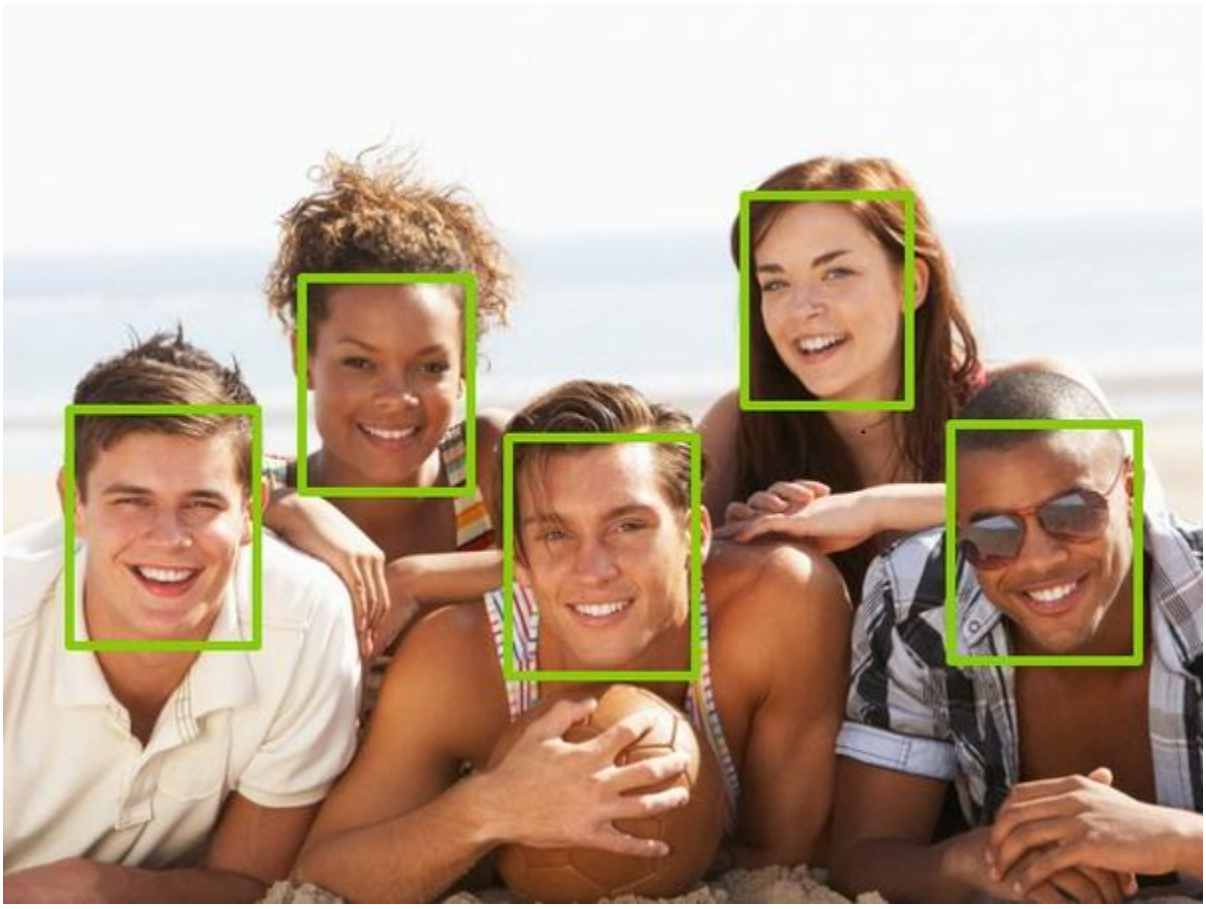

Figure 1 Face Detection

This approach may work well with well-defined sensitive objects, otherwise, the object detection based approach can no longer work. If there is no obvious characteristics of object, it will be hard to detect. The advantages for object detection is lots of implementations for object detection are used broadly. We can have these experiences as reference.

Another method is predefining the sensitive area, and it is a non-object related approach. In this method, the user will predefine the sensitive area or filter, which will be reserved by a software or platform for further operations.

To be specific, we will use QR code to pre-define sensitive area, and then to implement real time processing. QR code Accommodates up to 1850 capital letters or 2710 digits or 1108 bytes. It has strong capability of error tolerance, with error correction function. Besides, it's high reliable in decoding and easy to be made. The example of pre-defined sensitive area by QR code is showing as figure 2 below.

n nation's first high-speed railway, two government sources involved in making
id.

giants have both sent envoys to lobby Indonesian officials over the past two
veetening the terms of their bid for the contract worth about $5 billion.

Analysts believe that whoever wins will likely become the front runner for other high-speed rai projects coming up in Asia over the coming years, including one linking Kuala Lumpur and Singapore.

A cabinet-level committee led by chief economic minister Darmin Nasution will meet on Monday to make its recommendation on which country should build the rail line between the capital, Jakarta, and textile hub of Bandung.

President Joko Widodo is expected to announce the winner within days.

"Indonesia is leaning towards China because their proposal is less financially burdensome on the Indonesian government and because the issue of safety has been adequately addressed," a government source told Reuters.

A second government source said Indonesia wanted to strike a balance between the two powers in handing out high-profile infrastructure projects. Japan already holds contracts to build Jakarta's mass rapid transit system and the biggest coal-fired power plant in the region.

Both sources declined to be identified due to the diplomatic sensitivity of the issue for Jakarta and the economic stakes. Japan is Indonesia's second-largest investor, while China is its top trading partner.

"We have two partners and it will be good if we can maintain both of them. We have to be smart when taking this decision," Luky Eko Wuryanto, Indonesia's deputy minister of infrastructure and regional development, told Reuters last Friday after meeting China's ambassador to Indonesia.

The 150-km rail line should to cut the journey between Jakarta and Bandung to 35 minutes, from about three hours. Trains are expected to reach speeds of more than 300 km per hour.

Indonesia hopes to extend the line later to connect Jakarta with the city of Surabaya.

Japanese Prime Minister Shinzo Abe last week sent an envoy to offer a more attractive deal ahead of Monday's deadline, which was Japan's second revision in two weeks.

Japan's late manoeuvring brought protests from China, which said it was unfair Japan had been allowed to submit a new offer so close to the deadline, a minister said.

Figure 2 QR code predefined area

By using this method, the accuracy of recognition is dependent on the gratuity and clearance of QR code, thus we have to tradeoff either efficiency or recognition accuracy due to such disadvantages.

The approach of offline leakage preventing will also be implemented. The detection of QR code and image process are separated by two. A QR code detector is implemented independently. After the detection has been completed successfully, the result of detection will be reserved for the picture processor.

The advantage of this method is easy to implement and come with high efficiency, while it's not real time as an obvious disadvantage, and it also increases the operations for users.

As each one of these approaches has its advantages and disadvantages, in order to evaluate these three methods in sensitive information protection, the implementation and performance evaluation are applied and compared in this thesis.

3.2 Overview of proposed approach

In this thesis work, both none object based approach and the object based approach was implemented, and their performances were also compared.

The QR code visual marker in none object-based approach is composed of three main components, QR code generator, picture analyzer and picture processor. The QR code is generated by the first component including the coordinates, width and length. Thus in further step, the QR code can be printed out or integrated to the picture. Thus the user can use such sticker to prevent the accidental privacy leakage by scanning the QR code.

The second component, picture analyzer, will recognize the QR code, while the third component picture processor will remove or blur the sensitive area of the original picture.

In an object-based approach, the user need not to stipulate the QR code, the implemented application will recognize the face automatically, and then the same as none object-based approach, the picture processor will remove or blur the detected faces to protect the privacy of user.

3.3 Approach implementation

3.3.1 None object-based approach

For none object based approach, we have two implementations, including real time QR code visual marker and off line visual marker.

Regarding to real time QR code visual marker, it usually has three important components which are QR code generator, picture analyzer, and the QR code recognition and decoding. Usually QR code recognition and decoding are implemented as a picture processor, which will blur and encrypt the picture contains sensitive information.

For the QR code generator, there are two methods were proposed in this thesis for user to choose. The first one is Manual QR Code Generator which asks user to input the x, y, width and length. While the second one is Auto QR code Generator, the user only needs to choose the zone where contains sensitive information in the specific picture.

For offline method, the same QR coder generator of the real time visual marker was used. Since the accuracy of QR code detection highly depends on the diversity of internal structure, resolution ratio, scale, distortion and noise, it cannot be guaranteed easily. Therefore, the detection of QR code and image process are separated to improve the accuracy. In this manner, QR code detector is implemented independently, and we can adjust the pose of our hands until the beep sound which indicates the detection has been succeeded. Then the result of detection will be reserved for the picture processor. By using this method, we can improve the accuracy of QR code detection apparently, but it loses some efficiency because one step is separated by two.

3.3.2 Object-based approach

For object based approach, the face detection was used to demonstrate the concept, to be specific, the face was predefined as the object that needs to protect. The Visual Marker will encrypt it automatically, if any suspectable face was detected.

Chapter 4 Implementation and Evaluation

In this thesis work, the implementation of visual marker was illustrated and the scenario of using the visual marker is given, at the last the extension of visual marker application is finalized. All of our applications were based on Android platform, which means they could be used in version compatible android devices. With regards to none object based approach, QR was mainly used code to maximum the flexibility. The method by using QR code is more likely to be used in picture that captured from stable scenario.

4.1 Technical Background

QR code (short Quick Response Code) is a two-dimensional matrix type of barcode, which designed by the automotive industry in Japan and has gained popularity due to its large storage capacity and fast readability.

QR code is now used in much more widely scenarios, though it was originally designed for automotive industry only, including commercial tracking, traffic ticket, website URL (Uniform Resource Locator), and identification and so on.

It can store multi-purpose data type, and its basic architecture is showing as figure 3 below.

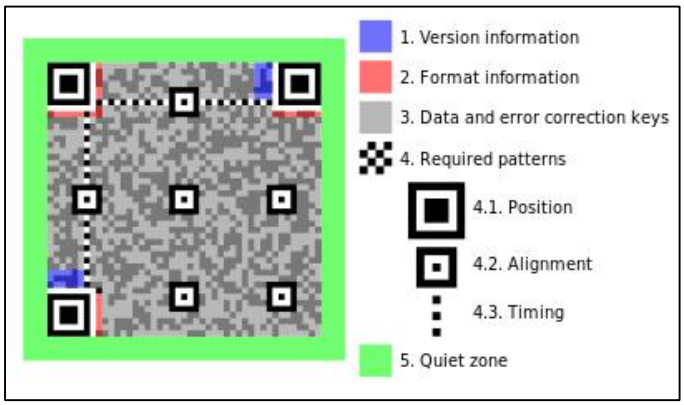

Figure 3 Sample of QR code

QR code has four advantages as below.

1. QR code can contain feasible and abundant information up to 1850 capital letters or 2710 digits or 1108 bytes;
2. Strong capability of error tolerance, with error correction function, which can restore 30% data for the maximum error correction level;
3. High reliability of decoding;
4. Easy to be generate;
5. Small printout size.

Lots of mature technologies of encoding and decoding information by using QR code are more convenient and reliable than using other kinds of code, to carry encrypted information, as QR code is a widely used standard now.

In the implementation of QR code, the information of sensitive privacy was encoded into QR code to maximum the flexibility of defining sensitive object.

Position and alignment are used for locating two-dimensional code of each QR code, their locations are fixed, but the size may vary. This is also named as FIP (Finder Pattern), because QR code is located and distinguished by the pattern at three corners of QR codes.

Format information contains two parts: one is the error correction level and another one is the mask pattern used for the symbol. Masking is used to break up patterns in the data area that might confuse a scanner, such as large blank areas or misleading features that look like the locator marks. The mask patterns are defined on a grid that is repeated as necessary to cover the whole symbol.

Version information illustrates the version two-dimensional code, and QR code has 40 kinds of version (usually black and white), from 21x21 (version 1), to 177x177 (version 40). Each version of the code has 4 more modules than the previous version in each side.

Data and error correction keys show the actual information saved by two-dimensional code, and the error correction information for correcting errors is caused by damage of the two-dimensional code. However, the quiet zone is the blank area reserved.

As the QR code can encode alphanumeric characters, a rich set of information can be fetched, thus we can extend and augment the content of information easily by using QR code.

ZXing Library in Android is used for the implementation of QR code generation and recognition. ZXing is a library which can support decoding and generating of barcodes (e.g. QR Code, PDF 417, EAN, UPC, Aztec, Data Matrix, Codabar) within images. The implementation instantiates the Java based barcode reader and generator library ZXing. Before running the program, we need to download the ZXing library to android studio by using the script of Gradle.

4.2 Difficulties in implementation

4.2.1 QR detection

Regarding to the example which raised before, in the company, the generated QR code can be used to stipulate an area used where contains sensitive information. All the participants are only allowed to use Visual Marker to take a picture of the whiteboard. After processing the picture by

this software, the sensitive privacy in the picture will be encrypted. In this way, we can provide the protection to these sensitive contents efficiently.

However, the direct detection of the QR code may not be an easy task, because of the internal structure of QR code are very diversified by different content. Besides, it's difficult to extract the features of the code area, as it will be greatly affected by resolution ratio, scale, distortion or noise.

To resolve this problem, two methods were proposed below.

In the first method, it is required to mark the area of QR code by black or red rectangle. But this also introduces new problem, the accuracy of the recognition will depend on the recognition of rectangles. Besides, there is possibility that the image has rectangle-like pattern, therefore, the accuracy cannot be guaranteed easily.

In the second method, recognition will be separated by two stages. The first step is detecting the QR code and getting the result, and then the Visual Marker will reserve the result for next stage. The second step is taking a picture of the object, and then process the picture by using the output of first stage.

4.2.2 QR code generation

Another difficulty in implementation is generating the QR code. The user can input the exact coordinates, width and heights directly to produce the QR code. But it's too ambiguous for normal user. For most users, they may have no concept about the exact digital number of the specified area.

Therefore, a flexible approach was proposed in this thesis to generate QR code by a scalable rectangle. The user can adjust it by draggling the four corners of the rectangle. After that, QR code will be generated by pressing the ensure button.

In the other implementation of face detector, we predefine the face is the object that we need to protect. If any susceptible face is detected, the Visual Marker will encrypt it automatically.

4.3 Real time QR code visual marker of none-object based approach

QR code visual marker is composed of three main components: QR code generator, picture analyzer and picture processor.

4.3.1 QR code generator

Two methods were provided to generate QR code.

1. The first approach is Manual QR Code Generator which asks user to input the coordinate x, y, as well as their desired width and length. Then visual marker will generate a QR code that includes these messages.

Regarding to the example scenario mentioned before, the audience wants to take a picture of the white board in a presentation, and meanwhile, they don't want to leak the important and private information. Therefore, the lecturer can use visual marker to generate a QR code, and then stick it in a corner of white board, to help all the participants to avoid the information leaking.

In this scenario, the lecturer needs to stipulate the range of the sensitive zone as shown in figure 4, then uses QR code visual marker to generate a QR code by indicating the coordinate X, Y, the width and the height.

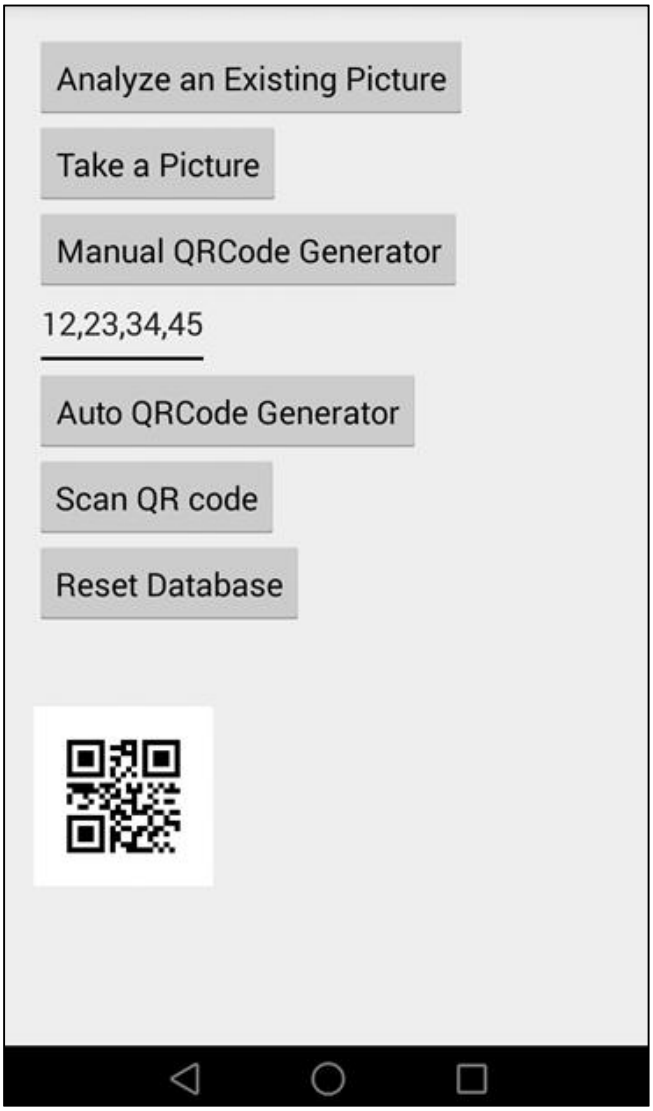

Figure 4 Example of QR code generator usage

The problem in such scenario needs to be resolved by figuring out how to locate the coordinates of target area in the picture. A real-time function was provided to adjust the area in a picture that need to be processed, and a QR code will be generated after choosing the target area. As long as the elements of target(x, y, width, length) are not changed, this QR code could be used at all time. By using this method, the user can get a better user experience.

2. Another approach is Automatic QR Code Generator, in which the user only needs to choose the zone that contains sensitive information in the specific picture. After pressing the button of Ensure, Generator will generate the QR code automatically.
The pseudocode of the QR code generation is shown as below.

```
//This object renders a QR Code as a BitMatrix 2D array of greyscale values

BitMatrix bitMatrix = new QRCodeWriter().encode(edit.getText().toString(),
BarcodeFormat.QR_CODE, width, height, hints);



for y=1 to y = height {

      for x=1 to x = width {

            if (bitMatrix.get(x, y)) {

                  set pixels [y * width + x] = black

            } else {

                  set pixels [y * width + x] = white

            }

      }

}
```

Then the user can choose the area that contains the sensitive information by moving and zooming in or out to match the area as they want. The detailed process of the application is to draw four small rectangles in the lower left, upper left; lower right, upper right vertices of the canvas. The color of these rectangles is labeled by red.

If user touches any one of these rectangle, the event will be processed, and the function onTouchEvent would be overridden. In the processing function of touch event, some judgements are taken to avoid the incorrect operations e.g. the range of the chosen area can't beyond the original range of picture.

When the event equals to MotionEvent.ACTION_DOWN, the coordinate x and y will be recorded, the color of the vertex rectangle will also be set to green. Once the event equals to MotionEvent.ACTION_MOVE, get the range by comparing with existed chosen area. If the area is changed, the coordinates of chosen area will be updated accordingly.

Now the lecturer can use QR code automatic generator to generate the QR code as shown in figure 5.

Figure 5 Example of generated QR code and target area

After the QR code is generated, user could reserve it for further use, e.g.: print it out, or stick it to any paper or object.

4.3.2 Picture analyzer

The most important purpose of the picture analyzer is to detect and analyze the QR code from the previous step as described in Chapter 4.3.1.

QR code detection and decoding information, which related to uncontrolled aspects such as QR code clarity, depends on the customer's cameras. In this manner, we can analyze an existed picture with QR code when the user taking the picture by themselves through the camera.

Based on the research of all current QR code catchers, we solved this problem by separating the recognition into several steps.

First of all, the QR code will be recognized just like other code catchers, and make the QR code as clear as possible, which could improve the ratio of recognition to almost 100%. Then, the information will be fetched from this QR code and stored inside.

Now, the picture can be processed with QR code as described in Chapter 4.3.1, after getting the QR code information from last step. The primary task in this step is to decode the QR code in the picture, and then to abstract the information inside, such as coordinate X, Y, and the width and length of the QR code.

By using the camera function of visual marker, our software will show an UI for user to capture the QR code. Refer to the figure 6 as below.

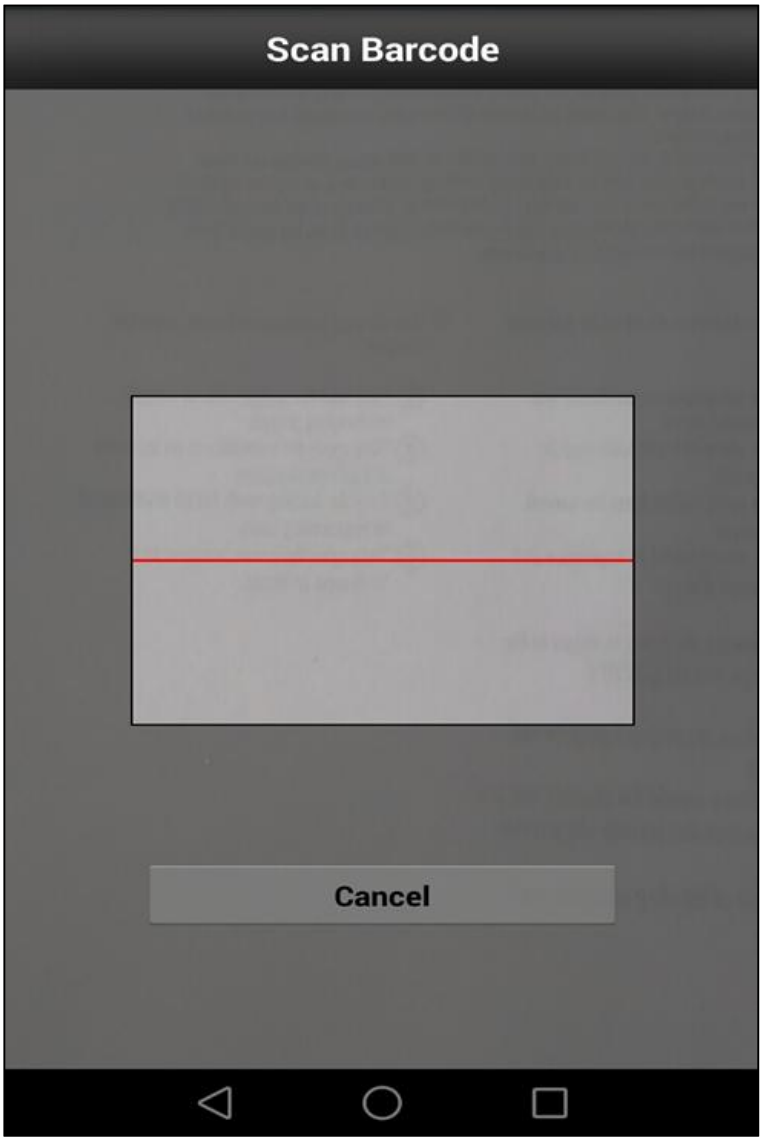

Figure 6 QR code scanner

If a complete and clear QR code is captured, the visual marker will recognize the QR code automatically, and a beep sound will be generated as a notification. The previous UI screen will be exited after that process, and result will be shown in the main user interface. Refer to figure 7.

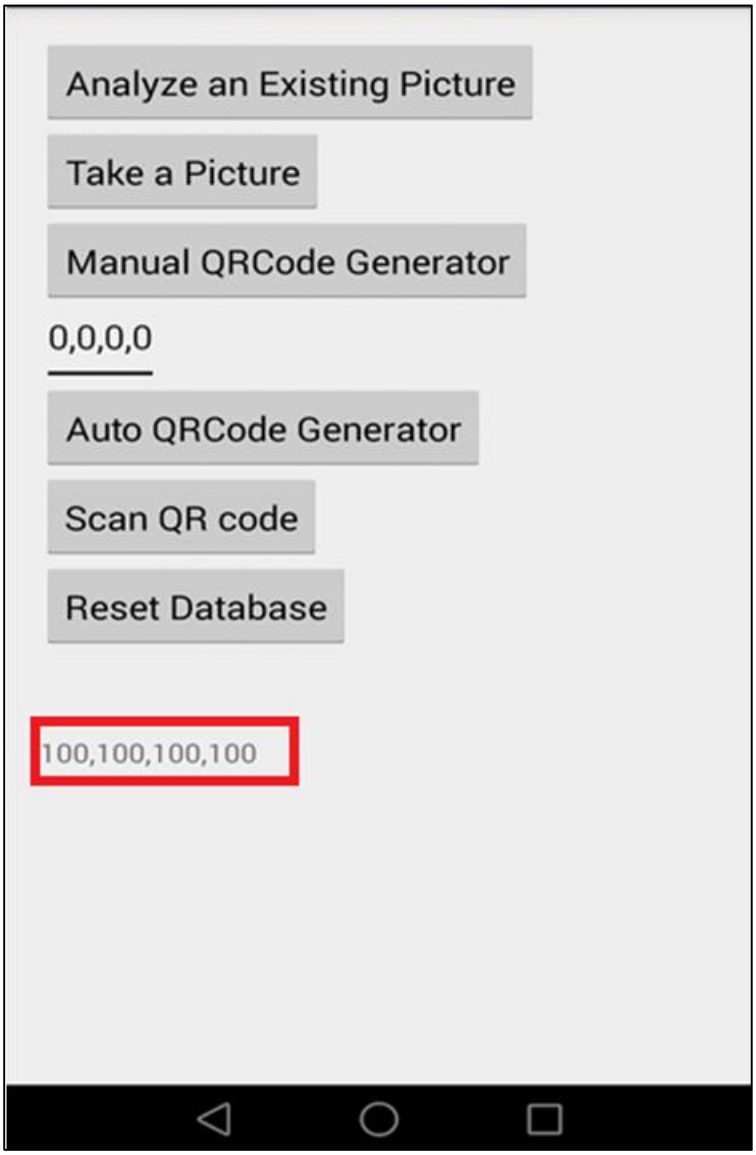

Figure 7 Result of QR code scanner

After that, the implementation will start to blur the corresponding area to avoid privacy leakage.

4.3.3 Picture processor

The picture processor is responsible for removing the sensitive area by encoding it, before that, the target area with coordinate X, Y and the width and length of the area are required to be calculated.

The QR code visual marker also takes responsibility for handling blurring information, and the degree of blurring depends on the authority of user input. Here an example was implemented to illustrate how QR code visual marker could be used to realize the protection of personal information.

By using this QR code, all the pictures taken by visual marker can recognize the QR code and then encrypt the area of sensitive information. Refer to figure 8, now the participant could record most content without leaking the privacy.

Figure 8 Example of picture processing

4.4 Offline visual marker of none-object based approach

The offline approach is separated into two steps, QR code recognition and image processing. A specific QR code recognizer was implemented as shown in figure 9.

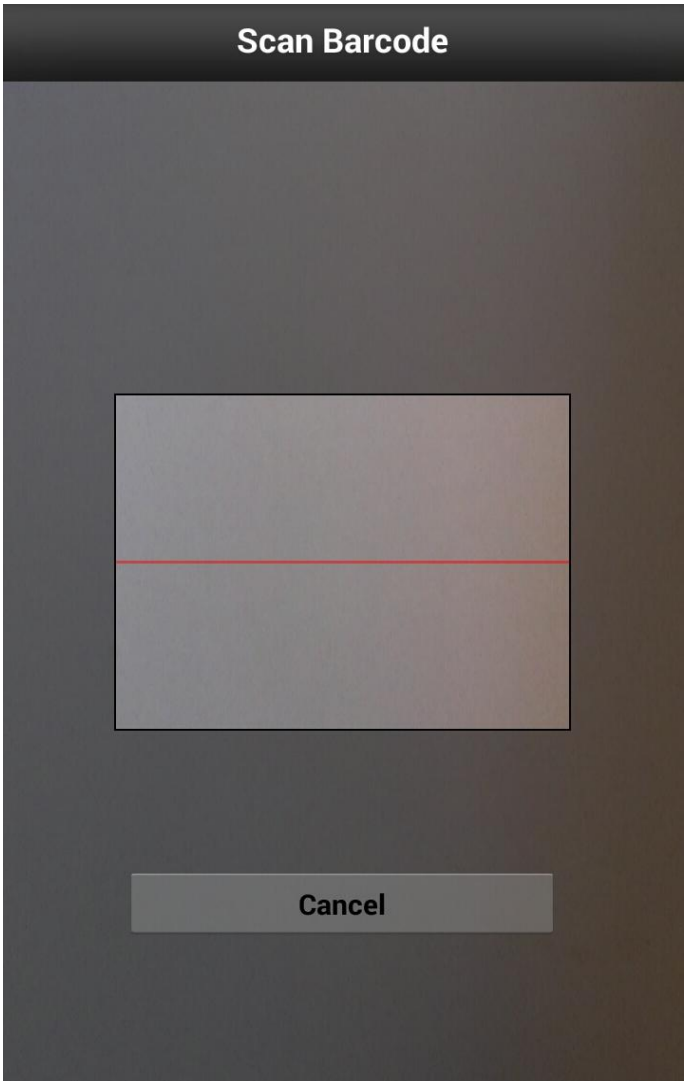

Figure 9 Example of QR code recognizer

Once the QR code was detected, a beep sound will be heard and the main UI will show the recognized result, which can be used in the next step of image processing.

4.5 Face detection of object-based approach

The face detection is used as an object-based approach, to implement our design in this thesis work. Face detection is a computer vision technology that determines the locations and sizes of human faces in arbitrary (digital) images. It detects facial features and ignores anything else, such as buildings, trees and bodies.

Face detection can be regarded as a more general case of face localization. In face localization, the task is to find the locations and sizes of a known number of faces. We mainly use OpenCV to realize the face detection. OpenCV is an open source framework and it can perform the face detection reliably and it's easy to implement. The build-in face detector can achieve 90% to 95% accuracy by rough estimation. However, the precondition for detection is the completion and clarity of face, if the face is lack of lightness and has some angle in head pose, the accuracy is hard to be guaranteed. Refer to [14].

A classifier (namely a cascade of boosted classifiers working with Haar-like features) stipulates a few hundred sample views of a specific object (e.g. a face or a tree), which also called positive examples. These examples are scaled to the same size, and negative examples - arbitrary images of the same size.

The algorithm in OpenCV is currently using the following Haar-like features which are the input to the basic classifiers, refer to figure 10.

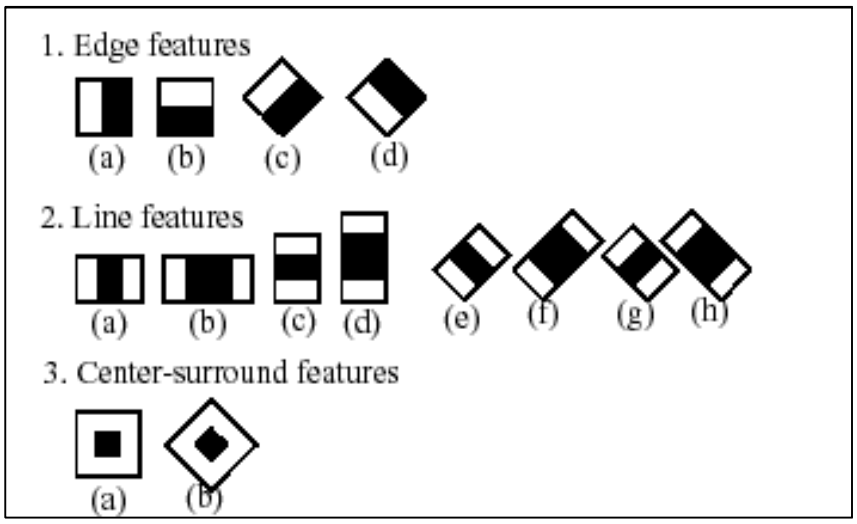

Figure 10 Haar-like features used in OpenCV algorithm

The classifier stipulates the features by its shape (e.g. 1a, 1b, etc.), position, the region of interest and the scale. For example, given a picture, all features will be extracted and compared with these prototypes. In the case of the second line feature (2b), the feature is calculated as the difference between the pixels in target and in image under the rectangle covering the whole feature (including the two white stripes and the black stripe). Using integral images can calculate the sums of pixel values under the rectangle.

The classifiers group these features in different stages instead of applying all types of features on a window. Usually, the first few stages only contain few numbers of features. The process will go to the next stage if the input sample is not matched by the features in the first stage. The remaining features on it will not been considered. Otherwise, if the comparison passed, the process of the second stage with other features will be executed continuously. Refer to figure 11 below.

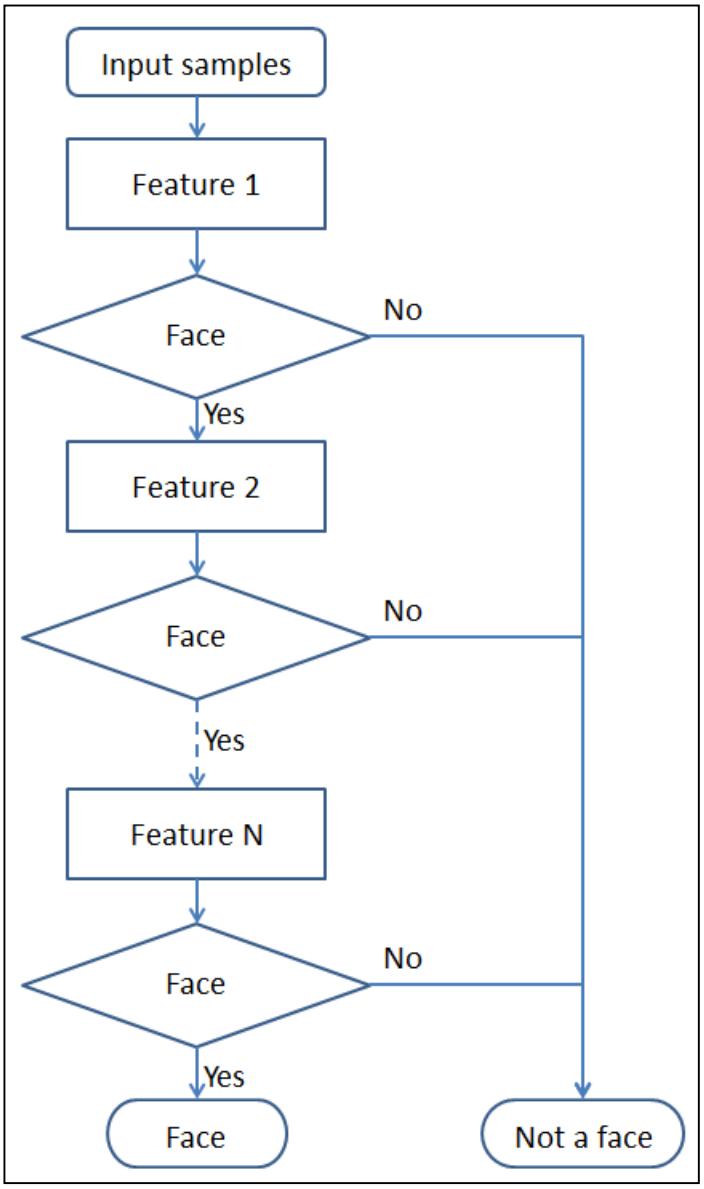

Figure 11 **Flow of face detection with different features**

The pseudocode of the face detection step by using OpenCV is shown as below.

```
//Load the required XML classifier

File mCascadeFile = new File(cascadeDir, "haarcascade_frontalface_alt2.xml");



CascadeClassifier faceDetector = new CascadeClassifier(mCascadeFile.getAbsolutePath());
```

```
//Load input image in Mat format
Utils.bitmapToMat(oriPicture, testMat);


//Use faceDetector.detectMultiScale() to find faces
public  void detectMultiScale(Mat image, MatOfRect objects);
```

In the last function detectMultiScale(Mat image, MatOfRect objects), the variable image indicates a matrix of the type <code>CV_8U</code>, which is representing an image where objects are detected. At the meanwhile, variable objects indicate a vector of rectangles where each rectangle contains the detected object.

In addition, the implementation above could detect multiple objects at the same time which means different faces can be detected and processed in one procedure.

The output will be marked by rectangles, by which the corresponding area including the faces will be encrypted for avoiding the privacy leakage, and will also be encrypted by the same method as QR code detection, refer to figure 12.

Figure 12 Example of face detection

Regarding to the encryption and decryption, we used AES (Advanced Encryption Standard) algorithm, which is a very popular and highly efficient symmetric algorithm. Based on the Rijndael cipher, which developed by two Belgian cryptographers, AES has been adopted by the U.S. government and is now used worldwide.

We used the seed algorithm to generate the seed for the AES, as android has evolved multiple versions, and the version over 4.2 is different from the previous version. Therefore, we changed the implementation to adapt this feature.

```
if (android.os.Build.VERSION.SDK_INT >=  17)

{
   sr = SecureRandom.getInstance("SHA1PRNG", "Crypto");
} else

{
   sr = SecureRandom.getInstance("SHA1PRNG");
}
```

The following procedure shows how to encrypt the data by using the class of Cipher in android. A key of 128 bits will be reserved for the encryption, while another key will be produced by using this key.

```
SecretKeySpec skeySpec = new SecretKeySpec(raw, "AES");
```

In order to create a Cipher object, the static function getInstance() in the class Cipher will be invoked by passing the name of the requested transformation as parameter to the application. And then initialize it by setting the mode to encryption mode and specifying the key.

```
Cipher cipher = Cipher.getInstance("AES");

cipher.init(Cipher.ENCRYPT_MODE, skeySpec);



//Last step, finish a multi-part transformation

byte[] encrypted = cipher.doFinal(clear);
```

The counter part of encryption is decryption, which has similar processes like encryption, such as getting the key, getting the instance, setting the mode (set mode to DECRYPT_MODE), as well as invoking the final step to finish the decryption as shown in the following pseudocode.

```
SecretKeySpec skeySpec = new SecretKeySpec(raw, "AES");


Cipher cipher = Cipher.getInstance("AES");
cipher.init(Cipher.DECRYPT_MODE, skeySpec);


byte[] decrypted = cipher.doFinal(encrypted);
```

4.6 Improvement of visual marker

In order to protect the user's privacy with guarantee of the flexibility the authority of access control for QR code is considered in this thesis work.

For instance, we can assign three authority levels for users as listed below.
1. Level 1 - Lowest level, user can't revert back any marked sensitive area;
2. Level 2 - Medium level, user can revert back few sensitive areas;
3. Level 3 - Highest level, user can revert back all sensitive areas.

We added a login UI functionality to implement this authority control, and predefined the access control level for user, refer to figure 13. Also, we encrypted the access control level to the QR code, which can couple permission granting with user actions within this software.

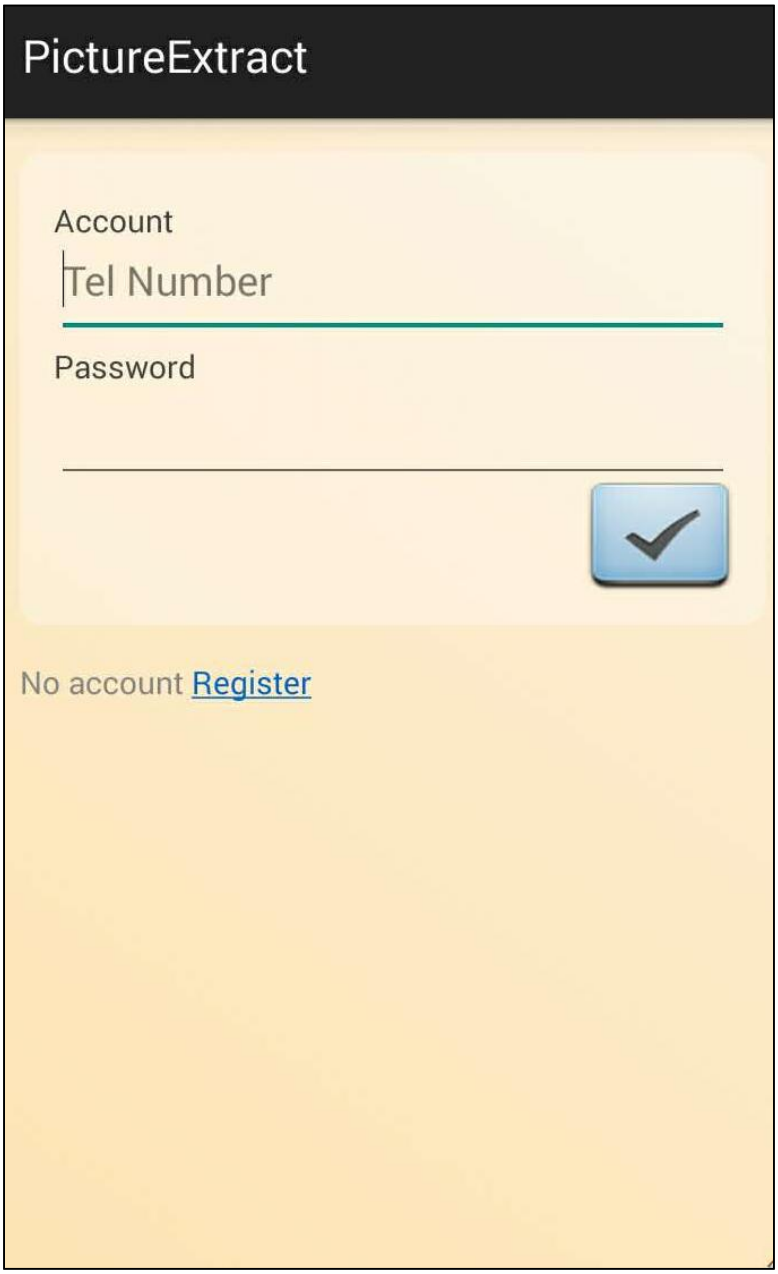

Figure 13 User login screen

For the user who has authority of level 1, the marked sensitive area cannot be reverted once the picture was encrypted, refer to figure 14.

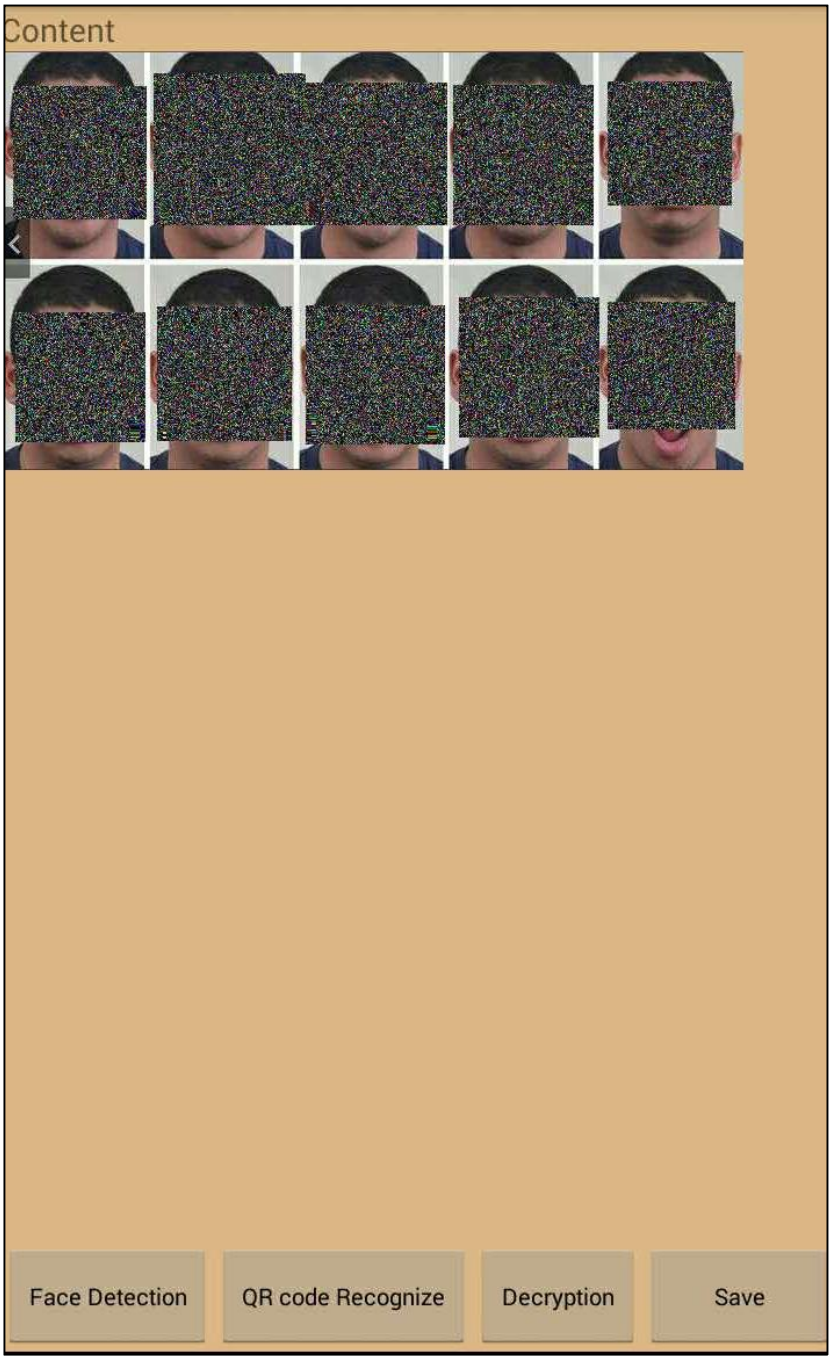

Figure 14 Example of encrypted image

For the user who has authority of level 2, only part of the encrypted areas can be reverted as shown in figure 15.

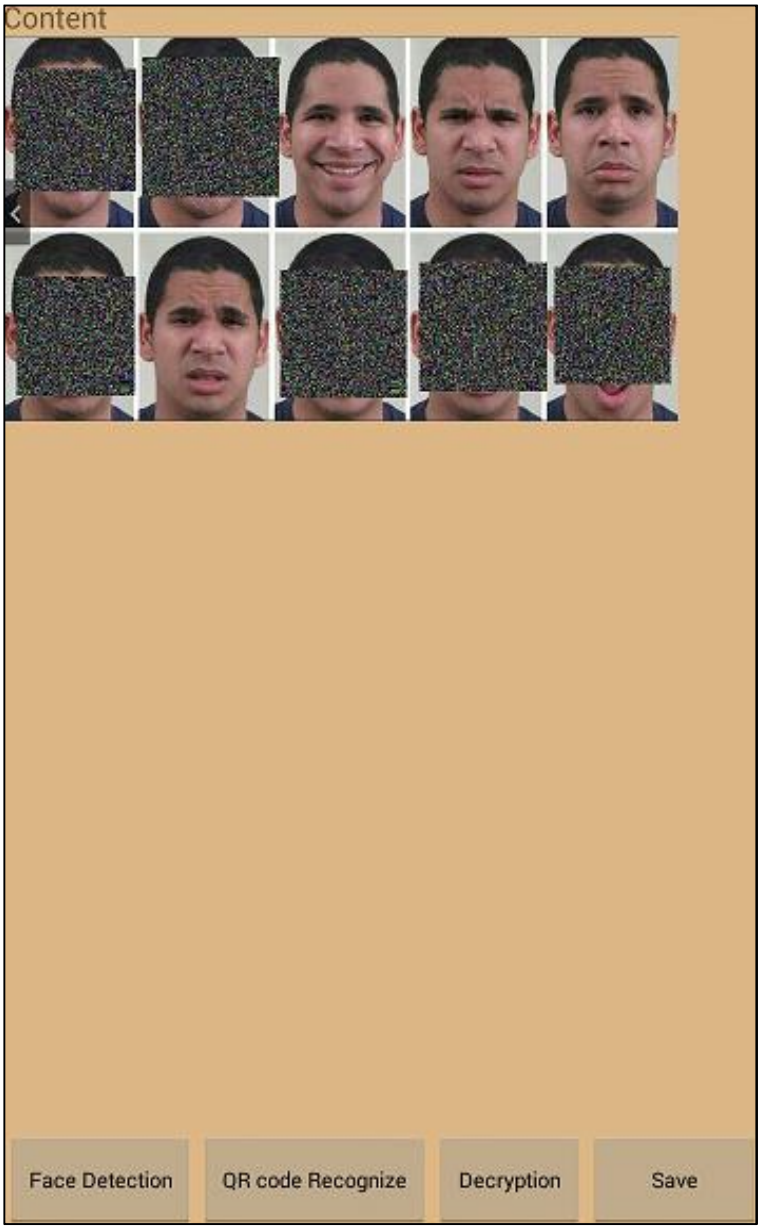

Figure 15 Example of partly decrypted image

For the user who has authority of level 3, all areas can be decrypted as shown in figure 16 below.

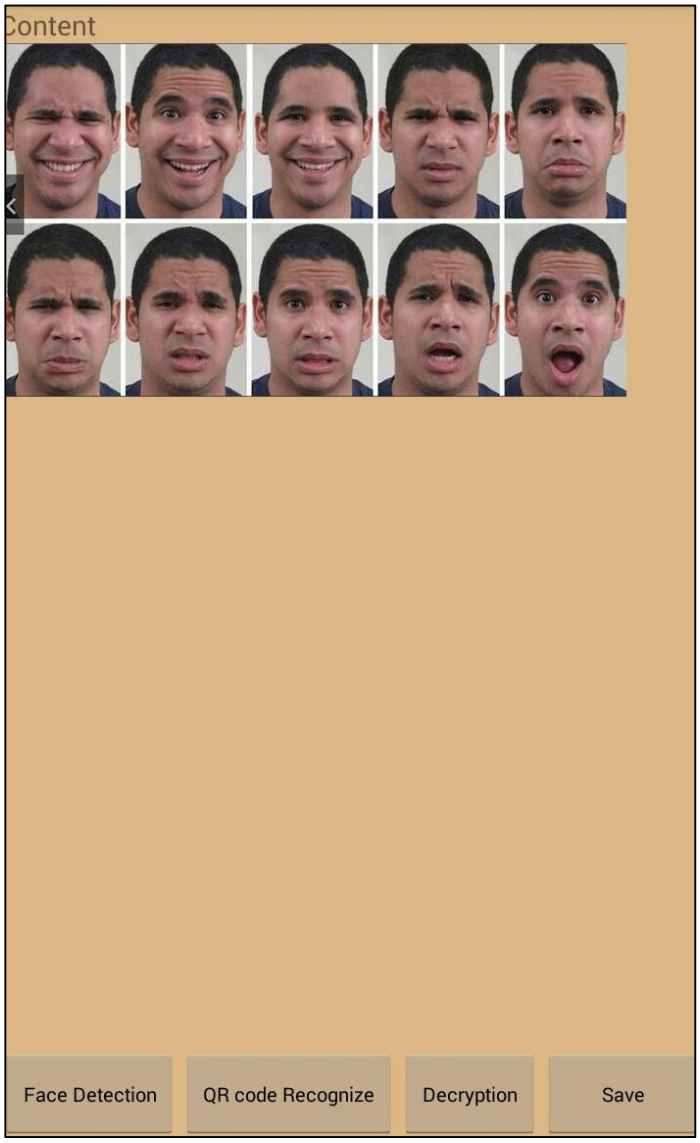

Figure 16 Example of fully decrypted image

Encryption and decryption were also implemented in this process, which related to the authority when removing it, therefore, part of or all the pictures could be reverted for further usage only if the user had higher authority.

A simple and efficient method was used to decode and encode the picture by generating a group of keys, which were used to decode or encode the pixels of the pictures.

To be specific, OpenCV was used to implement the functions of QR code recognition, and the QR code was encapsulated into a rectangle by the visual marker, thus the first step of recognition is to detect the rectangle of the picture.

1. Find edges in an image by using the Canny algorithm.

```
Imgproc.Canny(Mat image,Mat edges,double threshold1,double threshold2);
```

In this constructor function, the first parameter should be a single-channel 8-bit input image, which is the output of the function Imgproc.cvtColor(). Between the parameters threshold1 and threshold2, the smaller value will be used to link the edge, while the bigger value will be used to find the initial segment of the strong margin.

Usually the Canny algorithm contains five steps as listed below.
(1) Gaussian filter is applied to smooth the image
(2) Retrieve the intensity gradient
(3) Use non-maximum suppression
(4) Use double threshold to determine possible edges
(5) Track edge by hysteresis

2. Find a polygonal curve(s) with the specified precision.

```
Imgproc.approxPolyDP (InputArray curve, OutputArray approxCurve, double epsilon,
bool closed);
```

The function Imgproc.approxPolyDP()using Algorithm Douglas-Peucker can approximate a curve or polygon with few vertices and specified precision, then the distance between them is less or equal to the specified precision polygon.

4.7 Performance evaluation

The application was developed in Android Studio 1.4, which installed on Lenovo ThinkPad T430, the programming language is Java.

To test the functionalities and evaluate the performance, the application was eventually deployed on Huawei Mate 2 – L05 mobile device, on which Android 4.4.2 was used.

4.7.1 Real time QR code visual marker

100 times tests were made for each component and the statistical result data in Table 1 as shown below suspectable face was detected.

As QR code generator, encryption and decryption are always work fine, and they are not associated an accuracy rate, they were marked as Not Applicable (NA) in our test result.

Table 1 Statistical data of QR code visual marker

Component	Performance (Time in Milliseconds)
QR code generator	29
QR code recognition	929
Encryption	820
Decryption	578
Total	2356

4.7.2 Offline visual marker

The offline approach separated the recognition of QR code recognition, and reused the generator of QR code. The recognition sacrifices the time efficiency to improve the accuracy. Table 2 shows the difference in QR code recognition.

Table 2 Statistical data of offline visual marker

Component	Performance (Time in Milliseconds)
QR code recognition	1940.7
Encryption	799
Decryption	566
Total	3305

4.7.3 Face detection

Face detection included the time performance of detection, encryption and decryption， as shown in Table 3 below.

Table 3 Statistical data of face detection

Component	Performance (Time in Milliseconds)
Face detection	1949
Encryption	954
Decryption	167
Total	3070

4.7.4 Accuracy summary

The accuracy of the detection for each approach, as shown in Table 4 below.

Table 4 Statistical data of accuracy

Component	Accuracy (%)
QR code recognition(real-time)	95 (Depends on clarity of QR code)
QR code recognition(off-line)	99.5
Face detection	95 (Depends on clarity of face)

Chapter 5 Further Research

QR code could contain a maximum of 1108 bytes, which means it can contain lots of list of sensitive filters. Therefore, the area can be extended from one to several areas, such as, the multiple faces detection.

Beside QR code, there are other kinds of label can also be used to mark the object.
In order to expand the ability of passive UHF RFID precise localization, and increase the spatial resolution of RF in untrusted and uncontrolled environment, Alanson and his colleague from University of Washington proposed a new approach by combining the sensing capability of the localization methods. They designed a platform called WISP (Wireless Identification and Sensing Platform) which can locate tagged objects with millimeter accuracy optically by both computer system and human.

Continuous monitoring of some sensor data such as video or audio data, become more and more important on modern applications, but some privacy security issues might be brought in by continuous sensing in untrusted applications. Franziska and researchers from University of Washington and Microsoft Research designed a framework on different continuous sensing platforms to solve this problem. Their approach could resolve the problem of the end-user's permission management, by mediating access at the granularity of real-world objects rather than the whole continuous sensor data [16].

Thus, we can also extend our implementation to video in future. As the content of QR code is fixed in a period, so the quality of video processing could be guaranteed in the corresponding period.

Compared with certain context information, there exists some uncertain data. In order to extract the partial information from the presence of uncertainty, Vibhor, Dan and Evan formalized a natural semantics and proposed an algorithm to enforce such semantics, by using a provably optimal perturbation function. Meanwhile, they proposed another output perturbation algorithm to implement policies of access control by their own defined access control language which so called UCAL [15].

As the diversity of application, giving different permission to users by their privileges can also be introduced to increase the adaptability of scenarios.

For instance, the user with the highest privilege can decrypt and access to all the original pictures, while the user with the lowest privilege can only see the encrypted pictures without the authority in editing. The privilege of each user can be stored in the server database of network, and specific pin code or password is required asked when user try to decrypt these pictures have been processed by our implementation.

The solution of future research should improve the trade-off between the recognition speed and the accuracy of the state-of-the-art QR code detector, as the wearing devices usually have limited battery capacity and limited processing capability.

Moreover, self-study face recognition systems can improve its accuracy and flexibility. Different deep learning methods can be implemented by using new network architectures and learning algorithms on face related applications. However, the computing capability is still limited in single device, and one possible solution for this issue is to transfer the computing from device to cloud. Even so, this solution may also bring a new problem, due to the bandwidth of network. The latency of communication between client and data server shall be considered.

The idea of cloud computing is applicable when response time requires micro seconds. However, with regard to the application which asks milliseconds, it’s still difficult to achieve the desired time of the method.

Chapter 6 Conclusion

This thesis proposed the approach of visual markers that makes users to decide arbitrary sensitive objects in the real world. The face detection was also applied to predefine the sensitive area as well. In addition, the implementations of QR code visual marker/face detection and image encryption/decryption were presented to explain how the approach can be used to prevent the privacy leakage.

As mentioned in Chapter 5, lots of challenges need to be solved in future. For instance, most wearing devices have limited memory, processing unit, so we need to take the efficiency into consideration. However, as the development of technology and innovation, the capability of chips and processing increases gradually. We believe that all these obstacles can be fixed in near future, and more safety and security application, frame and platform will be proposed in wearing devices in near future.

Reference

1. L Ding On the Canny edge detector (2001) Pattern Recognition Volume 34, Issue 3, March 2001, Pages 721–725

2. M. Visvalingam and J. D. Whyatt. The Douglas-Peucker Algorithm for Line Simplification: Re-evaluation through Visualization(2007) Computer Graphics Forum 9(3): 213 – 225, October 2007

3. Belussi, L. F., and Hirata, N. S. Fast component-based qr code detection in arbitrarily acquired images: J. Math. Imaging Vis (2013) 45: 277. doi: 10.1007/s10851-012-0355-x

4. Enev, M., Jung, J., Bo, L., Ren, X., and Kohno, T. Sensorsift: Balancing sensor data privacy and utility in automated face understanding: ACSAC (2012)

5. Jana, S., Molnar, D., Moshchuk, A., Dunn, A., Livshits, B., Wang, H. J., and Ofek, E. Enabling Fine-Grained Permissions for Augmented Reality Applications with Recognizers: USENIX Security (2013)

6. Suman Jana, Arvind Narayanany Vitaly Shmatikov. A Scanner Darkly: Protecting User Privacy from Perceptual Applications: IEEE Symposium on Security and Privacy (Oakland) (2013)

7. Kalal, Z., Mikolajczyk, K., and Matas, J. Tracking learning detection: PAMI (2012)

8. Roesner, F., Kohno, T., and Molnar, D. Security and privacy for augmented reality systems: Commun. ACM (2014)

9. Schiff, J., Meingast, M., Mulligan, D.K., Sastry, S. and Goldberg, K.Y. Respectful cameras: Detecting visual markers in real-time to address privacy concerns. Protecting Privacy in Video Surveillance pp 65-89 (2009)

10. OpenCV Haar Feature-based Cascade Classifier for Object Detection http://docs.opencv.org/trunk/d7/d8b/tutorial_py_face_detection.html

11. Roesner, F., Molnar, D., Moshchuk, A., Kohno, T., and Wang, H. J. World-driven access control for continuous sensing: Tech. Rep. MSR-TR-2014-67, 2014

12. Ha, K., Chen, Z., Hu, W., Richter, W., Pillai, P., and Satyanarayanan, M. Towards wearable cognitive assistance: MobiSys (2014)

13. Algorithm Douglas-Peucker
http://en.wikipedia.org/wiki/Ramer-Douglas-Peucker_algorithm

14. Shervin EMAMI1, Valentin Petruț SUCIU,Facial Recognition using OpenCV: Journal of Mobile, Embedded and Distributed Systems (JMEDS) ISSN: 2067 – 4074 (2012)

15. Rastogi, V., Suciu, D., and Welbourne, E. Access control over uncertain data. VLDB (2008)

16. Roesner, F., Molnar, D., Moshchuk, A., Kohno, T., and Wang, H. J. World-driven access control for continuous sensing: Tech. Rep. MSR-TR-2014-67, 2014

17. Templeman, R., Korayem, M., Crandall, D., and Kapadia, A. PlaceAvoider. Steering first-person cameras away from sensitive spaces: NDSS (2014)

18. Ding, Ren: The Price of Free: Privacy Leakage in Personalized Mobile In-App Ads: Institute for Information Security & Privacy Cybersecurity Lecture Series (2016)

19. Yuta Nakashima, Noboru Babaguchi ,and Jianping Fan: Automatically Protecting Privacy in Consumer Generated Videos Using Intended Human Object Detector: Proceedings of the 18th ACM international conference on Multimedia Pages 1135-1138 (2010)

20. Shawn McCann, Jim Reesman :Object Detection using Convolutional Neural Networks. Stanford University(2013)

21. Darren Pauli,
http://www.theregister.co.uk/2014/06/25/brace_yourselves_brazil_dill_in_world_cup_wifi_spill/ (2014)

22. Jasper Hamill,
http://www.theregister.co.uk/2014/02/04/super_bowl_becomes_super_balls_up_as_cbs_broadcasts_wifi_code_to_the_world/ (2014)

23. Tim Jones,https://www.eff.org/deeplinks/2010/05/facebook-privacy-promises (2010)

24. Douglas Bonderud ,https://securityintelligence.com/news/public-wi-fi-security-risks-well-known-yet-completely-ignored-business-users/(2016)